

BEBAN-BEBAN LISTRIK TERKONTROL MELALUI *MINIMUM SYSTEM* BERBASIS *PAYLOAD DATA HANDLING* BERBANTUAN MIKROKONTROLER

Arief Goeritno^{1*}, Sopyandi², Rakhmad Yatim³

Program Studi Teknik Elektro, Fakultas Teknik, Universitas Ibn Khaldun Bogor
Jalan Sholeh Iskandar km.2 Kedung Badak, Tanah Sareal, Kota Bogor 16164 INDONESIA

²Laboratorium Instrumentasi dan Otomasi, Program Studi Teknik Elektro

³Pusat Teknologi Satelit, Lembaga Penerbangan dan Antariksa Nasional
Jalan Cagak Satelit km.4, Rancabungur, Kabupaten Bogor 16310 Telepon: (0251) 8623010
Email: arief.goeritno@ft.uika-bogor.ac.id

Abstrak

Telah dilakukan pengontrolan terhadap beban-beban listrik berbantuan sistem pengontrolan berbasis *Payload Data Handling* berbantuan mikrokontroler, melalui (1) pengintegrasian rangkaian elektronika sistem pengontrolan ke sistem mikrokontroler dan (2) uji verifikasi dan validasi. Pengintegrasian dilakukan terhadap sejumlah rangkaian elektronika untuk deteksi nilai tegangan, penyesoran nilai arus, penggerak relai, (ii) pabrikasi board untuk sistem mikrokontroler, (iii) pengawatan sistem pengontrolan dan pembuatan catu daya, dan (iv) pemrograman terhadap mikrokontroler berbasis bahasa pemrograman *BasCom AVR*. Uji verifikasi berupa simulasi berbantuan aplikasi *Proteus* dan uji validasi berupa pemberian beban listrik pada sejumlah titik terkontrol dan pengiriman berbantuan protokol *Bluetooth* dan penampilan pada perangkat berbasis *Android*. Pembuatan board untuk rangkaian elektronika dan sistem mikrokontroler berbantuan program aplikasi *EAGLE*, dimana pengawatan pada sistem mikrokontroler *ATmega32* berupa sensor arus dan tegangan sebagai masukan pada mikrokontroler dan ditampilkan pada *LCD* berukuran 4x16 sebagai keluaran untuk tampilan hasil pengukuran dan penampilan pada perangkat berbasis *Android*. Pengoperasian sensor saat pengukuran, digunakan sumber tegangan 5 volt dc dan komunikasi sensor dengan satu jalur data yang digunakan untuk perintah pengalamatan, pengambilan, dan pengiriman data dilakukan oleh mikrokontroler. Pemrograman mikrokontroler *Atmega32* meliputi pembuatan diagram alir dan penulisan sintaks. Diagram alir meliputi delapan tahapan, yaitu: (i) konfigurasi pin, (ii) deklarasi variabel, (iii) deklarasi konstanta, (iv) inisialisasi, (v) program utama, (vi) tampilan sensor tegangan dan arus, (vii) ambil dan kirim data, dan (viii) keluaran. Konfigurasi pin merupakan penentuan pin yang digunakan sebagai masukan atau keluaran, untuk sensor tegangan dan arus. Uji verifikasi berupa pengujian terhadap program berbasis bahasa aplikasi *Bascom AVR* di-compile ke sistem simulasi untuk mikrokontroler yang terdapat pada *Proteus*. Kondisi sistem dengan pengontrolan: (i) jika daya stabil dalam suatu pemakaian, maka beban listrik tetap dalam kondisi "on" dan (ii) ketika suatu ruangan berlebih dalam pemakaian daya listrik, maka beban listrik berubah menjadi "off" secara otomatis untuk pembatasan pemakaian daya listrik. Uji validasi berupa pemberian beban listrik pada titik-titik terkontrol dengan hasil yang tertampil pada *LCD* dan tertampilkan juga pada layar perangkat berbasis *Android* yang terkirim melalui jalur protokol *Bluetooth*.

Kata kunci: beban listrik, mikrokontroler, minimum system berbasis *PDH*, perangkat berbasis *Android*, perangkat *bluetooth*.

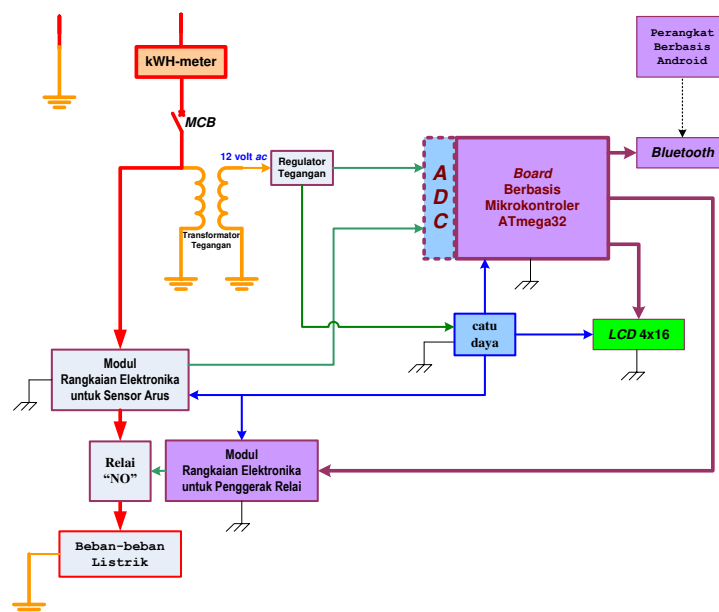
1. PENDAHULUAN

Penghematan dalam pemakaian energi listrik merupakan suatu keniscayaan. Manfaat penghematan energi sangat banyak, namun terdapat lima manfaat pokok, yaitu hemat dalam pengeluaran keuangan, terdapat peningkatan perekonomian, baik untuk lingkungan, peningkatan keamanan nasional, dan peningkatan kualitas hidup (**Alliance to Save Energy, 2012**). Berbagai cara dilakukan untuk penghematan tersebut. Salah satunya dengan pemasangan beban-beban listrik yang hemat energi. Terdapat cara lain untuk penghematan dalam pemakaian energi listrik, yaitu dengan pemasangan sistem elektronis (**Goeritno, 2016**) yang mampu deteksi pemakaian daya listrik melalui deteksi nilai tegangan dan arus yang diperoleh dari detektor tegangan dan sensor arus (**Linsley, 2005; EMSD, 2012**). Nilai tegangan dan arus sebagai masukan ke sistem elektronis, dimana sistem tersebut berfungsi juga untuk penyimpanan dan pengiriman informasi dalam hal penggunaan energi listrik (**Sopyandi, 2016**).

Pengintegrasian sistem dengan kriteria tersebut dikenal dengan sistem berbasis *Payload Data Handling* (Zhou, 2013).

Sistem berbasis *Payload Data Handling* (PDH), adalah sistem pemantauan, pengiriman, dan penerimaan data yang beroperasi secara otomatis melalui sistem pengontrolan jarak jauh atau dekat (Zhou, 2013). Keterkaitan sistem berbasis PDH dan pendeteksian pemakaian atau penggunaan daya listrik, meliputi (a) saluran pendistribusi daya listrik, (b) penggunaan saluran distribusi daya sebagai media pengiriman untuk keperluan komunikasi, dan (c) sistem pengontrolan berbasis PDH dengan pemanfaatan tegangan listrik (Sopyandi, 2016). Sistem berbasis PDH memiliki beberapa kelebihan, yaitu rancangan rangkaian yang ringkas, tidak perlu sumberdaya yang besar, dan terdapat tampilan untuk kemudahan pengguna dalam pemantauan beban yang sedang digunakan (Zhou, 2013). Hal ini dilakukan dalam kerangka antisipasi peningkatan beban listrik dan sumberdaya yang besar.

Berdasarkan uraian tersebut, dirancang bangun sebuah sistem pengontrolan berbasis PDH yang terdiri atas perangkat keras dan lunak. Perangkat keras berupa rangkaian elektronika berupa rangkaian sensor dan rangkaian pengontrol. Rangkaian sensor berfungsi untuk pendeteksian terhadap beban dalam hal pemakaian daya listrik. Rangkaian pengontrol terdiri atas konverter analog ke digital (*analog digital converter*, ADC) yang terdapat pada mikrokontroler ATmega32 (ATMEL Corporation, 2009) dengan bantuan bahasa pemrograman *BasCom AVR* (The MCS Electronics Team, 2008). Diagram blok sistem pengontrolann berbasis PDH berbantuan mikrokontroler, seperti ditunjukkan pada Gambar 1.



Gambar 1 Diagram blok sistem pengontrolan berbasis PDH berbantuan mikrokontroler

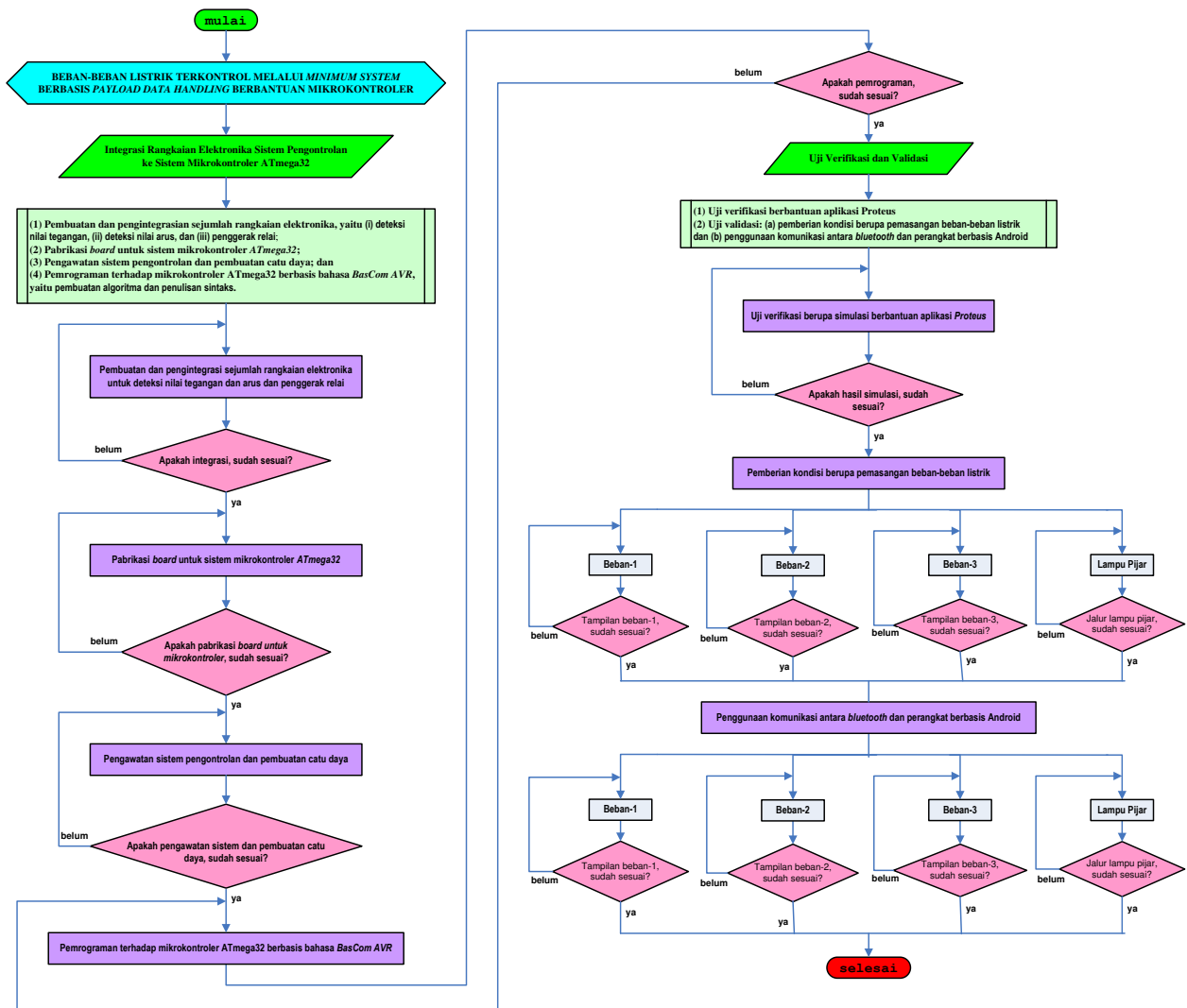
Berdasarkan Gambar 1 ditunjukkan, bahwa tujuan penelitian ini, yaitu: (1) memperoleh pengintegrasian rangkaian elektronika berbasis PDH ke sistem mikrokontroler ATmega32 dan pemrograman terhadap mikrokontroler ATmega32 (2) memperoleh uji verifikasi dan validasi sistem pengontrolan.

2. METODOLOGI

Untuk keperluan pelaksanaan metode penelitian, diperlukan bahan penelitian berupa: (i) sensor tegangan dan arus, (ii) *chip* AVR ATmega32, (iii) sejumlah modul relai (iv) sejumlah resistor, kapasitor, transistor, dan diode, (v) IC regulator 7805 (vi) *board* PCB, (vii) LCD 4x16, (viii) *transformator*, (ix) *downloader* mikrokontroler, dan (x) *Light Emiting Diode* (LED). Beberapa program aplikasi: (i) *Easily Applicable Graphical Layout Editor* atau EAGLE (Clark, 2008; CadSoft Computer, 2010; Aono, 2010), (ii) *Proteus* (Proteus2000, 1998), dan (iii) *ProgIsp* (AC-PG-USBASP USBASP AVR Programmer, 2008; AC-PG-USBASP USBASP AVR

Programmer, 2010; AC-PG-USBASP USBASP AVR Programmer, 2012). Bahasa pemrograman *BasCom AVR* (The MCS Electronics Team, 2008). Alat yang digunakan meliputi: (i) satu set *tools* elektronika, (ii) multimeter, (iii) setrika listrik, dan (iv) bor listrik/minidril.

Metode penelitian dibuat dalam bentuk diagram alir. Diagram alir penelitian, seperti ditunjukkan pada Gambar 2.



Gambar 2 Diagram alir metode penelitian

3. HASIL DAN BAHASAN

3.1 Integrasi Rangkaian Elektronika Sistem Pengontrolan ke Sistem Mikrokontroler ATmega32

Sebelum dilakukan pengintegrasian, terlebih dahulu dilakukan pabrikan setiap rangkaian elektronika dalam bentuk board untuk keperluan deteksi tegangan, penyensoran arus, modul penggerak relai; pembuatan board untuk mikrokontroler, pengawatan system pengontrolan dan pembuatan catu daya, dan pemrogramana berbasis bahasa *BasCom AVR*.

3.1.1 Rangkaian elektronika untuk detektor tegangan, sensor arus, dan penggerak relai

a) Rangkaian elektronika untuk deteksi tegangan

Diagram skematis rangkaian elektronika untuk deteksi tegangan, seperti ditunjukkan pada Gambar 3.

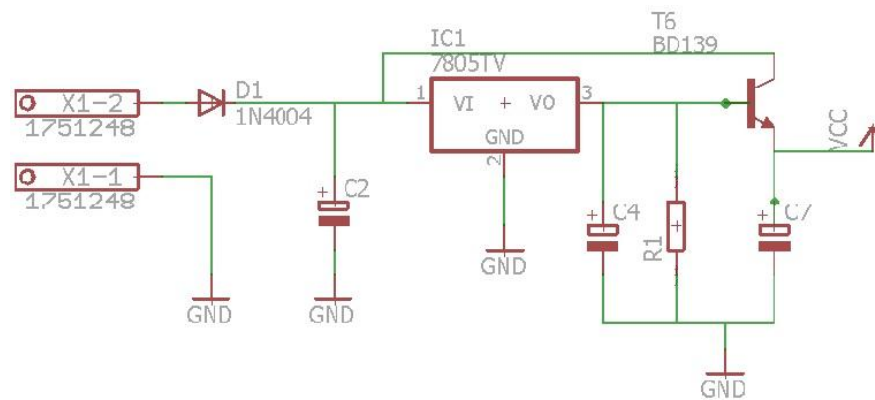
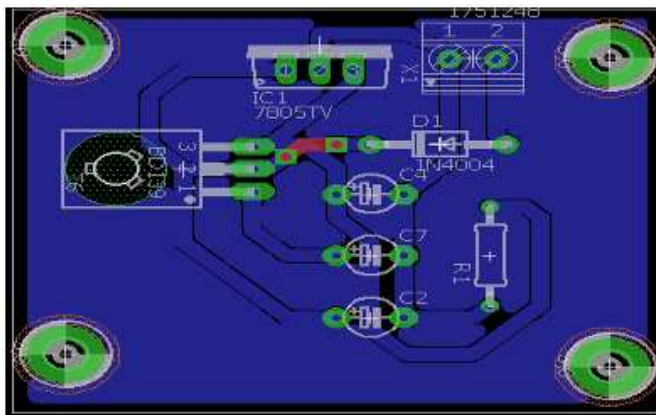


Diagram skematis rangkaian elektronika untuk deteksi nilai tegangan



Penampang *board* (PCB) untuk deteksi nilai tegangan



Bentuk fisis *board* untuk deteksi nilai tegangan

Gambar 3 Diagram skematis, penampang *board*, dan bentuk fisis *board* rangkaian elektronika untuk deteksi tegangan

Berdasarkan Gambar 3 ditunjukkan, bahwa diagram rangkaian elektronika tersebut dibuat dengan bantuan program aplikasi *EAGLE*, dimana dilakukan pembentukan jalur antar komponen, agar *pin-pin* yang terangkai pada *board* untuk deteksi tegangan dapat diketahui penggunaannya. Tata letak komponen merupakan langkah kedua untuk pengubahan diagram skematis menjadi bentuk rangkaian pada *board* (PCB) dengan perintah *switch to board* menggunakan program aplikasi *EAGLE*, dan langkah terakhir dicetak pada kertas foto dan dicetak pada PCB dengan pemanasan menggunakan setrika listrik. PCB berukuran 20 cm x 15 cm untuk penempatan sejumlah komponen elektronika. Komponen-komponen yang digunakan sesuai dengan kebutuhan dalam pembuatan modul deteksi tegangan, yaitu: diode 1N4001, resistor 330 kΩ, kapasitor 0,1 µF, 47 µF, 10 µF, transistor NPN BD139, dan IC regulator. Rangkaian elektronika tersebut terhubung ke sumber tegangan sebesar 220 volt dari pasokan listrik PLN.

b) Rangkaian elektronika untuk sensor arus

Diagram skematis rangkaian elektronika untuk sensor arus, seperti ditunjukkan pada Gambar 4.

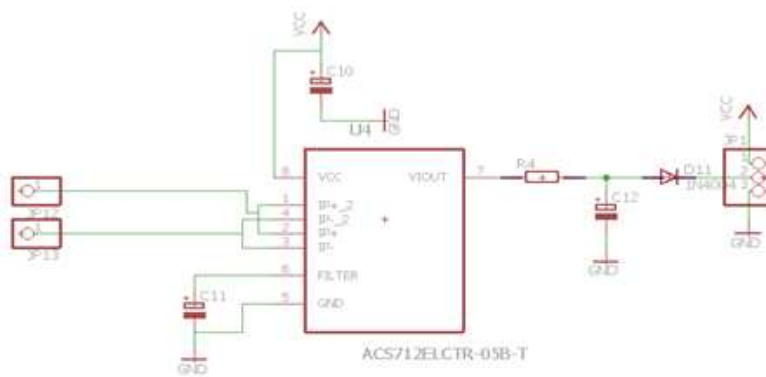


Diagram skematis rangkaian elektronika untuk sensor arus



Bentuk fisik modul untuk sensor arus

Gambar 4. Diagram skematis rangkaian elektronika untuk sensor nilai arus

Berdasarkan Gambar 4 ditunjukkan, bahwa diagram skematis rangkaian elektronika tersebut dibuat dengan bantuan program aplikasi *EAGLE*, dimana dilakukan pembentukan jalur antar komponen, agar *pin-pin* yang terangkai pada *board* sensor arus dapat diketahui penggunaannya. Tata letak komponen merupakan langkah kedua untuk pengubahan diagram skematis menjadi bentuk rangkaian pada *board* (*PCB*) dengan perintah *switch to board* menggunakan program aplikasi *EAGLE*, dan langkah terakhir dicetak pada kertas foto dan dicetak pada *PCB* dengan pemanasan menggunakan setrika listrik. Modul untuk sensor arus terdiri atas sejumlah komponen elektronik yang digunakan, yaitu diode 1N4004, kapasitor, dan sejumlah resistor 330 Ω dengan masukan yang diambil dari sumber tegangan 220 volt *ac* melalui konektor (*T-block*) dan keluaran rangkaian elektronika terhubung ke *port* masukan *Analog digital to convert* (*ADC*) pada mikrokontroler. Nilai keluaran sensor arus 100 mV/A untuk pembacaan pemakaian beban listrik yang ditampilkan di *LCD*.

c) Rangkaian elektronika untuk penggerak relai

Diagram skematis rangkaian elektronika untuk penggerak relai dan bentuk fisis modul relai, seperti ditunjukkan pada Gambar 5.

Berdasarkan Gambar 5 ditunjukkan, bahwa diagram skematis rangkaian elektronika tersebut dibuat dengan bantuan program aplikasi *EAGLE*, dimana dilakukan pembentukan jalur antar komponen, agar *pin-pin* yang terangkai pada *board* untuk relai dapat diketahui penggunaannya. Tata letak komponen merupakan langkah kedua untuk pengubahan diagram skematis menjadi bentuk rangkaian pada *board* (*PCB*) dengan perintah *switch to board* menggunakan program aplikasi *EAGLE*, dan langkah terakhir dicetak pada kertas foto dan dicetak pada *PCB* dengan pemanasan menggunakan setrika listrik. Modul terdiri atas konektor (*T-block*), relai, transistor, dan diode. Konektor yang terpasang di setiap kontak bantu (*auxiliary contact*) relai terhubung ke beban dan sensor arus ACS712. Koil relai dipasok dari catu daya dengan tegangan 12 volt *dc*. Setiap relai beroperasi setelah memperoleh sinyal dari mikrokontroler dan hasil pengoperasian dikirim ke mikrokontroler melalui kawat yang sama (*bidirectional*).

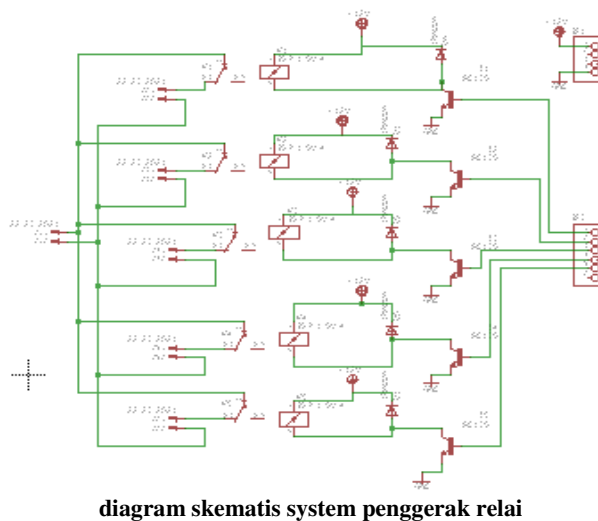


diagram skematis system penggerak relai

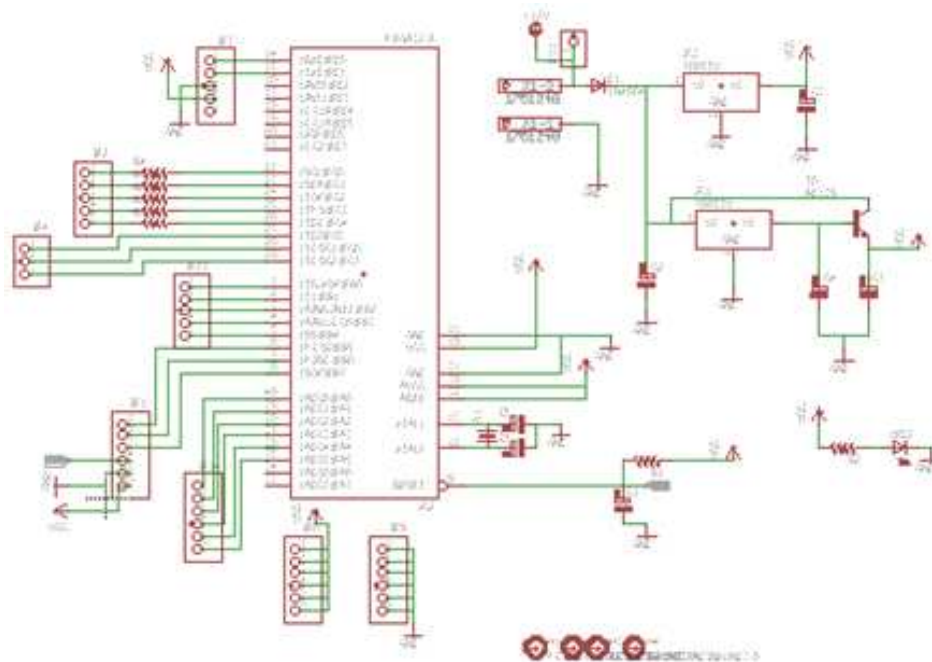


bentuk fisis modul relai

Gambar 5 Diagram skematis dan rangkaian elektronika untuk penggerak relai dan bentuk fisis modul relai

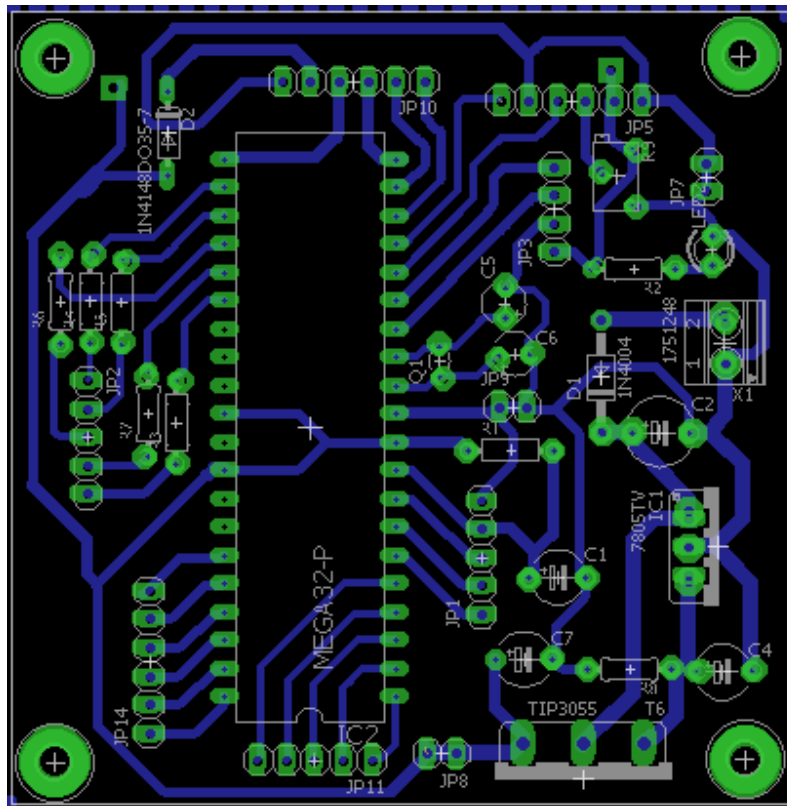
3.1.2 Board untuk sistem mikrokontroler ATmega32

Pembuatan board untuk sistem mikrokontroler ATmega32, meliputi: a) Diagram skematis rangkaian elektronika untuk sistem mikrokontroler ATmega32, b) Bentuk fisis *board* untuk sistem mikrokontroler ATmega32, c) Bentuk fisis *board* untuk sistem mikrokontroler ATmega32, dan d) seperti ditunjukkan pada Gambar 6.



Gambar 6 Diagram skematis rangkaian elektronika untuk sistem mikrokontroler ATmega32

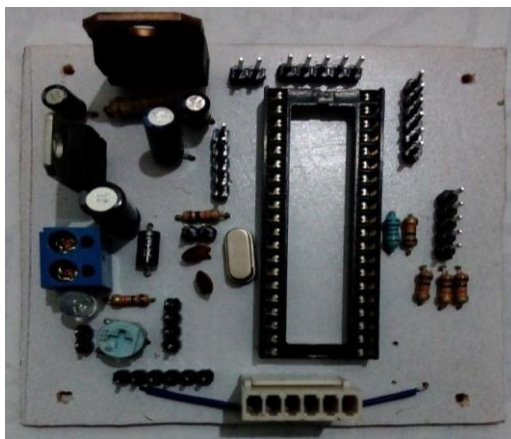
Berdasarkan Gambar 6 ditunjukkan, bahwa diagram skematis rangkaian elektronika tersebut dibuat dengan bantuan program aplikasi *EAGLE*, dimana dilakukan pembentukan jalur antar komponen, agar *pin-pin* yang terangkai pada *board* untuk relai dapat diketahui penggunaannya. Tata letak komponen merupakan langkah kedua untuk pengubahan diagram skematis menjadi bentuk rangkaian pada *board* (*PCB*) dengan perintah *switch to board* menggunakan program aplikasi *EAGLE*, dan langkah terakhir dicetak pada kertas foto dan dicetak pada *PCB* dengan pemanasan menggunakan setrika listrik. Bentuk fisis *board* untuk sistem mikrokontroler ATmega32, seperti ditunjukkan pada Gambar 7.



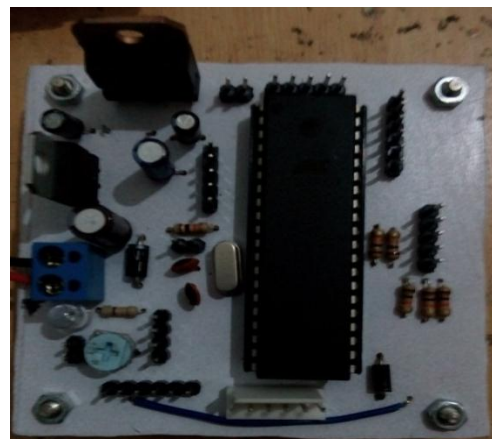
Gambar 7 Bentuk fisis *board* untuk sistem mikrokontroler ATmega32

b) Tata letak komponen-komponen pada *board*

Tampilan tata letak komponen-komponen pada *board* untuk sistem mikrokontroler ATmega32 dilakukan melalui tahapan pengeboran dan penyolderan kaki-kaki komponen. Bentuk fisis *board* untuk sistem mikrokontroler ATmega32, seperti ditunjukkan pada Gambar 8.



sebelum pemasangan *chip* ATmega32



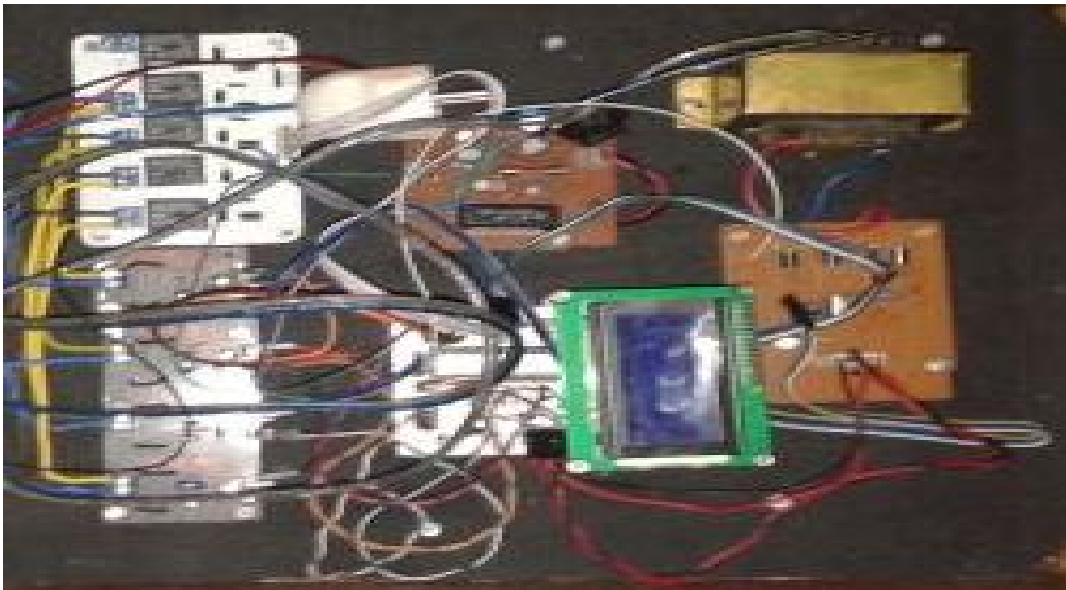
setelah pemasangan *chip* ATmega32

Gambar 8 Bentuk fisis *board* untuk sistem mikrokontroler ATmega32

Berdasarkan Gambar 8 ditunjukkan, bahwa *board* digunakan untuk komponen-komponen sesuai dengan kebutuhan yang meliputi *chip* mikrokontroler ATmega32, resistor variabel 10 k Ω , kapasitor 2,2 nF, osilator kristal untuk frekuensi 12 MHz., dan *pin-pin* untuk *header*.

3.1.3 Pengawatan sistem pengontrolan dan pembuatan catu daya

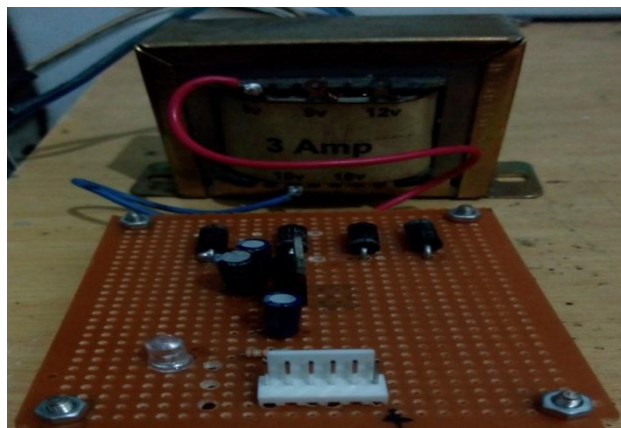
Pengawatan sistem pengontrolan secara keseluruhan, seperti ditunjukkan pada Gambar 10.



Gambar 10 Pengawatan sistem pengontrolan secara keseluruhan

Berdasarkan Gambar 10 ditunjukkan, bahwa *pin* serial data yang terhubung ke mikrokontroler memberikan perintah pengalamatan pada *pin* data sensor untuk pengukuran pendeteksian tegangan dan pengukuran arus. *Pin* yang digunakan pada modul detector dan sensor terletak pada *port-D*, dengan 8 *pin* yang tersedia hanya digunakan 5 *pin* saja, yaitu D0, D1, D2, D3, dan D4. Untuk sambungan ke *LCD*, terletak pada *port-C*, dengan 8 *pin* yang tersedia hanya digunakan 6 *pin*, yaitu C0, C2, C4, C5, C6, dan C7.

Catu daya sangat penting dalam pasokan daya untuk pengoperasian sejumlah komponen elektronika. Terdapat satu catu daya untuk keperluan mikrokontroler maupun komponen pendukung lain, yaitu penggunaan tranformator sistem tegangan sisi sekunder pada nilai 12 volt *ac*. Bentuk fisik catu daya, seperti ditunjukkan pada Gambar 9.



Gambar 9 Bentuk fisik catu daya

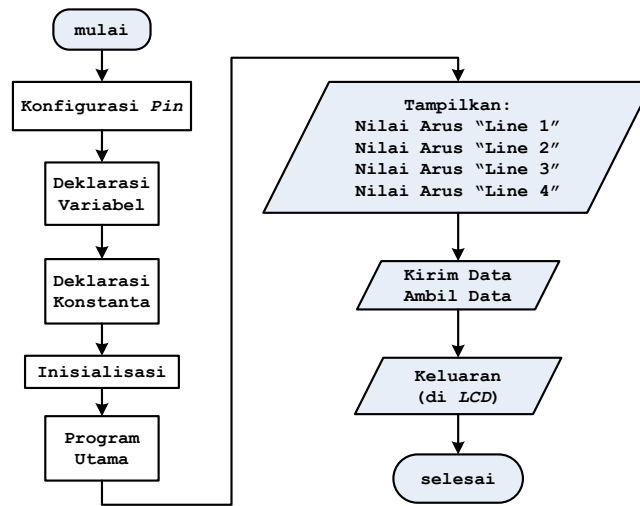
Berdasarkan Gambar 9 ditunjukkan, bahwa catu daya tersebut digunakan untuk pasokan daya ke sistem sensor-transduser, mikrokontroler ATmega32, relai, dan *LCD* 4x16. Hasil pengukuran terhadap catu daya 5 volt *dc* dari regulator diketahui, bahwa tegangan keluaran tanpa beban pada nilai rata-rata 4,9 volt *dc* dan tegangan keluaran dengan beban pada nilai rata-rata 4,8 volt *dc*, sehingga tegangan pada catu daya untuk mikrokontroler relatif konstan (stabil).

3.1.4 Pemrograman terhadap mikrokontroler ATmega32 berbasis bahasa *BasCom AVR*

Program yang digunakan pada prototipe sistem pendeteksian kelebihan daya listrik berbantuan detector tegangan dan sensor arus berbantuan mikrokontroler, yaitu bahasa *Basic Compiler (BasCom)*. Fungsi utama *BasCom* untuk meng-*compile* kode program menjadi *hexa-desimal* (bahasa mesin). Pemrograman terhadap mikrokontroler ATmega32 dilakukan melalui pembuatan algoritma dan penulisan sintaks yang diakhiri dengan uji verifikasi terhadap program yang telah dibuat ke dalam program aplikasi berbantuan bahasa aplikasi *Proteus*.

1) Algoritma

Pembuatan algoritma didasarkan kepada metode diagram alir (*flowchart*). Diagram alir pemrograman terhadap mikrokontroler ATmega32 secara keseluruhan, seperti ditunjukkan pada Gambar 11.



Gambar 11 Diagram alir pemrograman terhadap mikrokontroler ATmega32 secara keseluruhan

Berdasarkan Gambar 11 ditunjukkan, bahwa algoritma pemrograman terdiri atas delapan tahapan yaitu: i) konfigurasi *pin*, ii) deklarasi variabel, iii) deklarasi konstanta, iv) inisialisasi, v) program utama, vi) tampilkan nilai arus beban, vii) ambil dan kirim data, dan viii) keluaran.

2) Sintaks

Sintaks ditulis dalam bahasa *BasCom* mengikuti setiap tahapan pada masing-masing bagian algoritma.

i) Konfigurasi *pin*

Konfigurasi *pin* merupakan penentuan *pin* yang digunakan, baik sebagai masukan maupun keluaran. *Pin* tersebut akan dijadikan sebagai parameter dalam setiap pengalamatan program untuk penentuan *pin* pada ATmega32, baik untuk deteksi tegangan, sensor arus, penggerak relai, maupun *LCD 2x16*. Konfigurasi *pin* pada pemrograman untuk prototipe sistem pengontrolan arus beban listrik, yaitu:

```

Config Lcdpin = Pin
Db4 = Portb.4
Db5 = Portb.5
Db6 = Portb.6
Db7 = Portb.7
E = Portb.2
Rs = Portb.0
Config Adc = Pin
Adc1 = Portc.0
Adc2 = Portc.1
Adc3 = Portc.2
  
```

ii) Deklarasi variabel

Deklarasi variabel dilakukan untuk pendeklarasian jenis data yang harus dikerjakan. Deklarasi variabel pada pemrograman untuk sistem pengontrolan arus beban listrik, yaitu:

```
Start Adc
```

```
Dim Tegangan As Byte
```

```
    Dim Cont_sk1 As Byte , Cont_sk2 As Byte , Cont_sk3 As Byte ,  
    Cont_lamp As Byte,
```

```
    Dim Ask1 As Byte , Ask2 As Byte , Ask3 As Byte , Alamp As Byte
```

```
    Dim Sk1 As Single , Sk2 As Single , Sk3 As Single , Lamp As Single
```

iii) Deklarasi konstanta

Deklarasi konstanta merupakan pemberian nilai konstanta pada program berdasarkan *datasheet* dari sensor yang merupakan masukan pada sistem pengontrolan arus beban listrik berbasis mikrokontroler ATmega32. Deklarasi konstanta langsung disebutkan besar nilai. Deklarasi konstanta tidak menggunakan tanda titik dua (:) seperti pada deklarasi variabel, tetapi penggunaan tanda sama dengan (=).

```
Config Portc = Output
```

```
In_tegangan Alias Getadc(0)
```

```
In_sk1 Alias Getadc(1)
```

```
In_sk2 Alias Getadc(2)
```

```
In_sk3 Alias Getadc(3)
```

```
In_lamp Alias Getadc(4)
```

```
Pb_sk1 Alias Pind.3
```

```
Pb_sk2 Alias Pind.4
```

```
Pb_sk3 Alias Pind.5
```

```
Pb_lamp Alias Pind.6
```

```
Out_sk1 Alias Portc.3
```

```
Out_sk2 Alias Portc.4
```

```
Out_sk3 Alias Portc.5
```

```
Out_lamp Alias Portc.6
```

iv) Inisialisasi

Inisialisasi yaitu pemberian inisial terhadap program yang dibuat untuk mengetahui status dari setiap perintah pada program. Inisialisasi ini diharapkan dapat mempersingkat perintah pada program selanjutnya.

```
Locate 1 , 1
```

```
Lcd "power line"
```

```
Wait 3
```

```
Locate 2 , 1
```

```
Cls
```

```
Do
```

```
Ask1 = In_sk1
```

```
Ask2 = In_sk2
```

```
Ask3 = In_sk3
```

```
Alamp = In_lamp
```

```
If Pb_sk1 = 0 Then
```

```
    Cont_sk1 = Cont_sk1 + 1
```

```
Do
```

```
Loop Until Pb_sk1 = 1
```

```
Elseif Pb_sk2 = 0 Then
```

```
    Cont_sk2 = Cont_sk2 + 1
```

```
Do
```

```
Loop Until Pb_sk2 = 1
```

```
Elseif Pb_sk3 = 0 Then
```

```
    Cont_sk3 = Cont_sk3 + 1
```

```
Do
```

```
Loop Until Pb_sk3 = 1
```

```
Elseif Pb_lamp = 0 Then
Cont_lamp = Cont_lamp + 1
Do
Loop Until Pb_lamp =
```

v) Program utama

Program utama merupakan sumber dari pengontrolan program, karena semua perintah pada program diurutkan dari tampilan awal, pengambilan data, menampilkan data pada *LCD* dan reaksi atau *output* dari program yang dibuat.

```
Locate 1 , 1
Lcd "sk1= " ; Fusing(sk1 , "#.##") ; " Amp "
Locate 2 , 1
Lcd "sk2= " ; Fusing(sk2 , "#.##") ; " Amp "
Locate 3 , 1
Lcd "sk3= " ; Fusing(sk3 , "#.##") ; " Amp "
Locate 4 , 1
Lcd "lmp= " ; Fusing(lamp , "#.##") ; " Amp "
```

vi) Tampilkan nilai arus

Penampilan nilai tegangan dan arus dilakukan untuk mengetahui setiap perubahan yang terjadi pada suatu titik instalasi. Nilai arus ditampilkan dengan ketentuan, bahwa tampilan nilai dalam empat untuk masing-masing beban listrik.

vii) Ambil dan kirim data

Pengambilan data dilakukan sampai mendapatkan perubahan datanya, setelah itu maka data tersebut dikirim untuk selanjutnya ditampilkan pada *LCD*.

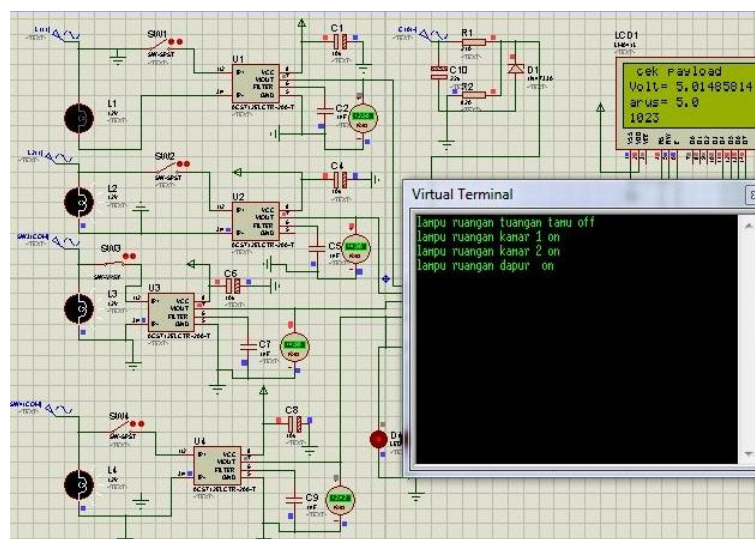
viii) Keluaran

Keluaran program merupakan reaksi yang diakibatkan dari sumber atau sensor untuk selanjutnya ditampilkan pada *LCD*.

3.2. Uji Verifikasi dan Validasi

3.2.1 Uji verifikasi

Uji verifikasi terhadap program berbasis bahasa *BasCom* dilakukan dengan bantuan program aplikasi *Proteus*. Rangkaian prototipe sistem pengontrolan terlebih dahulu dirangkai dengan bantuan program aplikasi *Proteus*, kemudian program yang sudah dibuat dengan bahasa *BasCom* di-*comfile* menjadi bentuk heksa atau bahasa mesin dan di-*download*-kan ke rangkaian tersebut. Tegangan dan arus yang terdeteksi pada simulasi dapat diatur sesuai dengan yang diinginkan untuk pembuktian, apakah program yang telah dibuat sudah sesuai dengan yang diharapkan atau belum. Tampilan pada uji verifikasi, seperti ditunjukkan pada Gambar 12.



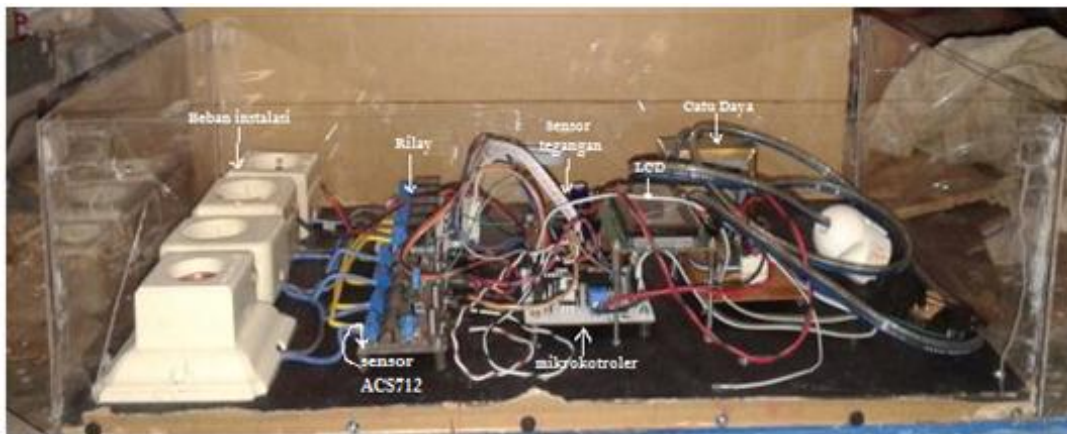
Gambar 12 Tampilan pada uji verifikasi

Berdasarkan Gambar 12 ditunjukkan, bahwa uji verifikasi terhadap program berbasis bahasa *BasCom* yang telah dibuat, dimana untuk sensor tegangan tergantung setelan pemberian tegangan masukan yang diberikan dan untuk nilai arus tergantung setelan pada resistor variabel dalam simulasi. Program yang telah dibuat dan telah disimulasikan sesuai hasil yang diharapkan.

3.2.2 Uji validasi

1) Pemberian kondisi berupa pemasangan beban-beban listrik

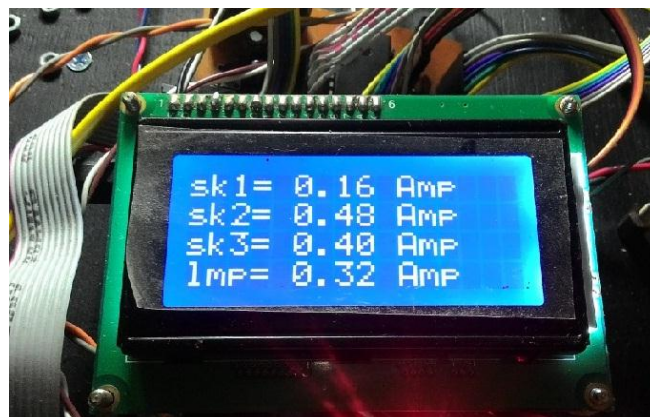
Prototipe sistem pengontrolan berbasis *PDH* berbantuan mikrokontroler berupa kotak berdimensi 50 cm x 30 cm x 15 cm untuk penempatan transformator, sistem berbantuan mikrokontroler, sensor-transduser, dan *LCD* 16x4. Pengukuran kinerja terhadap sistem dilakukan melalui pemberian kondisi buatan. Bentuk fisis prototipe sistem pengontrolan berbasis *PDH* berbantuan mikrokontroler untuk instalasi rumah tinggal, seperti ditunjukkan pada Gambar 13.



Gambar 13 Bentuk fisis prototipe sistem pengontrolan berbasis *PDH* berbantuan mikrokontroler untuk instalasi rumah tinggal

Perubahan beban terjadi, apabila nilai tegangan terdeteksi yang disertai perubahan nilai arus. Pengukuran tegangan dan arus dilakukan dengan beban listrik, sehingga sensor akan deteksi perubahan tegangan dan arus yang kemudian memberikan informasi nilai tegangan dan arus telah tercapai pada nilai tertentu. Nilai tegangan terukur dalam bentuk bilangan desimal sebagai konversi bilangan biner 8 *bit* dan nilai arus terukur ampere (A). Pembacaan hasil pengukuran, ditampilkan melalui *LCD* 4x16. Nilai tegangan dan arus terukur sesuai dengan keadaan yang terdeteksi. Untuk pemakaian beban yang melebihi batas, maka sensor arus mendeteksi.

Pengukuran kinerja sistem dilakukan terhadap beberapa contoh titik instalasi, meliputi 1) kotak kontak pertama, 2) kotak kontak kedua, dan 3) kotak kontak ketiga, dan 4) saklar terhubung ke lampu. Tampilan hasil pengukuran sejumlah beban listrik, seperti ditunjukkan pada Gambar 14.



Gambar 14 Tampilan hasil pengukuran sejumlah beban listrik

Beban terpasang pada kontak pertama berupa *solder*, bahwa hasil pengukuran terhadap beban dengan nilai arus sesuai kemampuan sensor, sebesar 0,16 ampere. Nilai arus terukur telah diubah dari nilai beban pemakaian, bernilai 4,19 volt (dalam bentuk nilai digital 8 bit). Untuk nilai arus yang terdapat dalam pemakaian beban, dimana hasil dari pengukuran sensor tersebut dipengaruhi nilai arus yang terdapat dalam pemakaian beban, maka kenaikan beban sebanding dengan nilai arus yang dideteksi. Beban terpasang pada kontak kedua berupa *charger* untuk *laptop*, bahwa hasil pengukuran terhadap beban dengan nilai arus sesuai kemampuan sensor, sebesar 0,4 ampere. Nilai arus terukur telah diubah dari nilai beban pemakaian, bernilai 4,47 volt (dalam bentuk nilai digital 8 bit). Untuk nilai arus yang terdapat dalam pemakaian beban, dimana hasil dari pengukuran sensor tersebut dipengaruhi nilai arus yang terdapat dalam pemakaian beban, maka kenaikan beban sebanding dengan nilai arus yang dideteksi. Beban terpasang pada kontak ketiga berupa kipas angin, bahwa hasil pengukuran terhadap beban dengan nilai arus sesuai kemampuan sensor, sebesar 0,24 ampere. Nilai arus terukur telah diubah dari nilai beban pemakaian, bernilai 4,35 volt (dalam bentuk nilai digital 8 bit). Untuk nilai arus yang terdapat dalam pemakaian beban, dimana hasil dari pengukuran sensor tersebut dipengaruhi nilai arus yang terdapat dalam pemakaian beban, maka kenaikan beban sebanding dengan nilai arus yang dideteksi. Beban terpasang berupa lampu, bahwa hasil pengukuran terhadap beban dengan nilai arus sesuai kemampuan sensor, sebesar 0,32 ampere. Nilai arus terukur telah diubah dari nilai beban pemakaian, bernilai 4,41 volt (dalam bentuk nilai digital 8 bit). Untuk nilai arus yang terdapat dalam pemakaian beban, dimana hasil dari pengukuran sensor tersebut dipengaruhi nilai arus yang terdapat dalam pemakaian beban, maka kenaikan beban sebanding dengan nilai arus yang dideteksi.

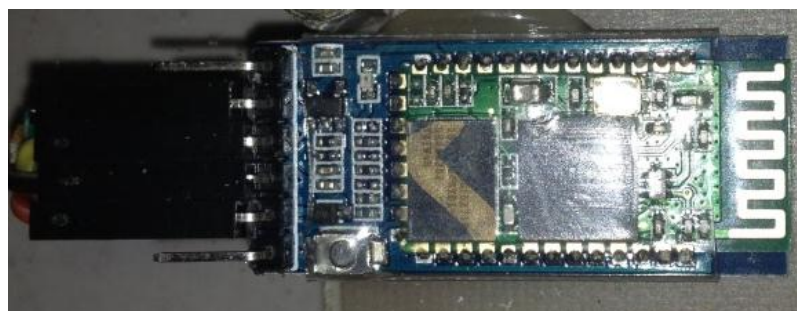
Berdasarkan tampilan nilai arus hasil pengukuran yang ditunjukkan pada *LCD* sebagai bentuk hasil pengukuran dari sistem pengontrolan berbasis *payload data handling* berbantuan mikrokontroler untuk instalasi listrik rumah tinggal. Tegangan terukur pada masing-masing terminal beban listrik sebesar 214 volt *ac*. Hasil pengukuran nilai arus yang dideteksi oleh sensor arus ACS712, seperti ditunjukkan pada Tabel 1.

Tabel 1 Hasil pengukuran nilai arus yang dideteksi oleh sensor arus ACS712

No.	Beban Listrik Terpasang	Nilai Arus (ampere)
1	<i>Solder</i>	0,16
2	<i>Charger</i> untuk <i>Laptop</i>	0,40
3	Kipas Angin	0,24
4	Lampu	0,32

2) Penggunaan komunikasi antara *bluetooth* dan piranti berbasis android

Aplikasi berbasis android dengan penggunaan koneksi *bluetooth HC-05* yang ditempatkan pada *board* mikrokontroler. Modul *bluetooth HC-05*, seperti ditunjukkan pada Gambar 15.



Gambar 15 Modul Bluetooth HC-05

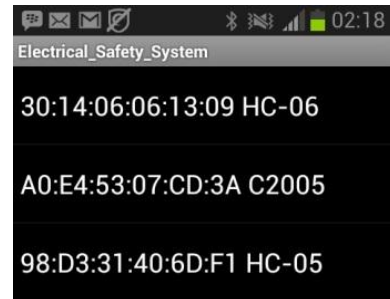
Berdasarkan Gambar 15 ditunjukkan, bahwa modul *bluetooth* terdapat 2 buah *pin* untuk tegangan 5 volt *dc* dan 2 buah *pin* untuk *RX* dan *TX*. Perangkat komunikasi berbasis Android dengan telepon genggam (*handphone*) merk Oppo R827 untuk pengoperasian aplikasi. Aplikasi berisi cara

perbaikan, untuk pemantauan pemakaian arus dalam saklar instalasi. Aplikasi ini dapat beroperasi, apabila dikoneksikan dengan modul *bluetooth HC-05*.

Tampilan tahapan penggunaan perangkat berbasis Android yang terhubung ke *Bluetooth*, seperti ditunjukkan pada Gambar 16.



Tampilan layar pertama saat aplikasi dioperasikan



Tampilan pencarian modul *bluetooth*



Tampilan layar *Connection* saat *bluetooth* sudah terhubung



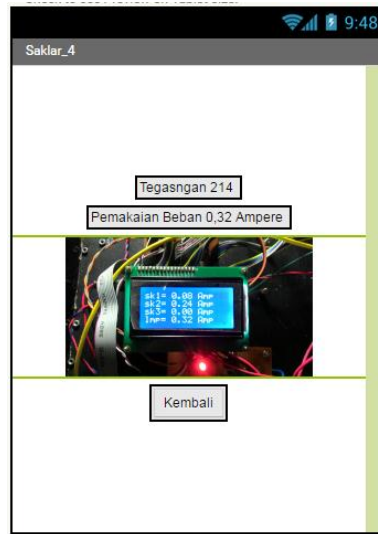
Tampilan layar *Maintenance*

Gambar 16 Tampilan tahapan penggunaan perangkat berbasis Android yang terhubung ke *bluetooth*

Berdasarkan Gambar 16 ditunjukkan, bahwa terdapat sejumlah tahapan. Tahapan pertama kali tampilan pada layar saat aplikasi dioperasikan. Aplikasi ini tidak terhubung pada *bluetooth*, dan merujuk pengguna untuk menghubungkan dengan *bluetooth HC-05* dengan cara menekan gambar *bluetooth* pada layar untuk pencarian *bluetooth HC-05*. adalah permintaan untuk penghubungan dengan modul *bluetooth HC-05*. Tahapan kedua pencarian *bluetooth HC-05* dapat beroperasi apabila *bluetooth* pada *handphone* harus dalam keadaan hidup. Dengan mengklik pilihan *HC-05* maka *bluetooth* terhubung dan merubah tampilan pada layar *Connection* menjadi “*Connected*” atau terhubung antara *bluetooth* pada *handphone* dan *bluetooth HC-05*. Tahapan ketiga keterhubungan *bluetooth* pada *handphone* dan *bluetooth HC-05* dapat difungsikan tombol “*Open*” pada layar. Fungsi tombol “*Open*” pada layar akan membuka layar “*Maintenance*” dan penekanan pada tombol “*Close App*” untuk fungsi penutupan aplikasi. Tahapan keempat untuk tindakan perbaikan apabila terjadi gangguan pada jalur instalasi pada rumah dan tombol “*Kembali*” untuk kembali ke layar *Connection*.

Tampilan pengontrolan pada sejumlah beban listrik, seperti ditunjukkan pada Gambar 17. Berdasarkan Gambar 17 ditunjukkan, bahwa pengontrolan pada jalur kotak kontak pertama, kedua, ketiga, dan keempat tertampilkan pada layar berupa “tampilan pemakaian beban” yang terkirim secara manual. Untuk kondisi dimana terjadi penggunaan beban yang tidak stabil dan tertampilkan

foto pada *LCD* yang terjadi gangguan, dan tombol “kembali” berfungsi untuk pengembalian ke layar *Maintenance*.



Gambar 17 Tampilan pengontrolan pada beban-beban listrik

4. KESIMPULAN

Berdasarkan hasil dan bahasan dapat disimpulkan, bahwa pembauatan prototipe sistem pengontrolan berbasis payload data handling berbantu mikrokontroller untuk instalasi listrik rumah tinggal telah sesuai dengan tujuan penelitian.

(1) Integrasi sistem ke sistem mikrokontroler ATmega32 ditunjukkan, bahwa diagram rangkaian skematis terlebih dahulu dibuat berbantuan program aplikasi *Eagle* untuk pembentukan jalur antar komponen, agar *pin* pada *board* dapat diketahui penggunaannya, baik *board* pada penyearah tegangan atau *board* mikrokontroler. Pembuatan *board* melalui tahapan penyetrikaan, pelarutan, dan pengeboran *board PCB* yang diakhiri dengan pemasangan komponen. *Board* untuk mikrokontroler ATmega32 menyediakan lima *port* masing-masing untuk konektor catu daya 5 volt *dc*, sensor, *LCD 4x16*, *downloader*, dan *output*. Lima *port* tersebut merupakan masukan (*input*) dan keluaran (*output*) yang berasal dari *pin* ATmega32. Terdapat satu catu daya untuk keperluan mikrokontroler maupun komponen pendukung lain, yaitu catu daya dengan sistem tegangan 12 volt *Ac* dengan arus 2 ampere. Hasil pengukuran terhadap catu daya 5 volt *dc* dari *regulator* diketahui, bahwa tegangan keluaran tanpa beban pada nilai rata-rata 4,9 volt *dc*, sedangkan tegangan keluaran dengan beban pada nilai rata-rata 4,8 volt *dc*. Pemrograman mikrokontroler ATmega32 untuk pengoperasian sistem, dilakukan penanaman program berbasis bahasa *BasCom* melalui delapan tahapan, yaitu: (i) konfigurasi *pin*, (ii) deklarasi variabel (peubah), (iii) deklarasi konstanta (tetapan), (iv) inisialisasi, (v) program utama, (vi) tampilan: nilai arus (line-1) sampai (line-4), (vii) ambil dan kirim data, dan (viii) keluaran.

(2) Hasil simulasi mendekati hasil yang diharapkan, untuk sensor arus yang tergantung setelah pemberian tegangan masukan yang diberikan, dan arus tergantung setelah dengan resistor variabel pada simulasi. Pengukuran terhadap kinerja sistem ditunjukkan, bahwa pemilihan beban listrik sangat penting, agar dapat diketahui nilai saat pengukuran dan tetap akurat, tidak berubah-ubah. Sistem pengontrolan berbasis *payload data handling* mampu menunjukkan arus beberapa contoh titik instalasi listrik sesuai kemampuan sensor. Salah satu nilai arus diatas nilai 0,40 atau setara dengan nilai tegangan 4,47 volt mengindikasikan, bahwa sistem pengontrolan terhadap contoh titik instalasi tersebut perlu dilakukan pengujian dan pengukuran nilai arus dapat dikatakan layak, apabila nilai tegangan pada kisaran 220 volt *ac*.

DAFTAR PUSTAKA

AC-PG-USBASP USBASP AVR Programmer, (2008), *User Guide Version 1.0*, AC-PG-USBASP-UG. [Online]. Available: <https://kanga.gerbilator.org/Microcontrollers/Atmel->

- AVR/Users%20Guide%20(AC-PG-USBASP-UG-V1.0).pdf, tanggal akses 11 September 2016.
- AC-PG-USBASP USBASP AVR Programmer, (2010), *User Guide Version 1.2*, AC-PG-USBASP-UG. [Online]. Available: [https://protostack.com.au/download/Users%20Guide%20\(AC-PG-USBASP-UG-V1.2\).pdf](https://protostack.com.au/download/Users%20Guide%20(AC-PG-USBASP-UG-V1.2).pdf), tanggal akses 11 September 2016.
- AC-PG-USBASP USBASP AVR Programmer, (2012), *User Guide Version 2.0*, AC-PG-USBASP-UG. [Online]. Available: <http://eecs.oregonstate.edu/education/docs/ece375/USBASP-UG.pdf>, tanggal akses 11 September 2016.
- Alliance to Save Energy, (2012, July, 20), Top 5 Reasons To Be Energy Efficient, [Online]. Available: <http://www.ase.org/resources/top-5-reasons-be-energy-efficient> (accessed: January, 30 2017).
- Aono, K., (2011), “*Application Note: PCB Design with EAGLE*”, ECE480 Design Team 5, Department of Electrical & Computer Engineering, Michigan State University, pp. 1-33.
- ATMEL Corporation, (2009, May), 8-bit Microcontroller with 32 Kbytes In-System Programmable Flash. [Online]. Available: <http://site.gravitech.us/Arduino/NANO30/ATMEGA328datasheet.pdf> (accessed: **January, 30 2017**).
- CadSoft Computer, (2010), “*Eagle Easily Applicable Graphical Layout Editor Manual Version 5*”, CadSoft Computer Inc., pp. 37-80.
- Clarke, T., (2008), “*The EAGLE Schematic & PCB Layout Editor - A Guide*”, Course Material, Department of Electrical & Electronic Engineering, Imperial Collage London, pp. 1-17.
- EMSD, (2012), *Code of Practice for Energy Efficiency of Building Service Installation*. Government of Hong Kong, pp. 1-53. [Online]. Available: http://www.beeo.emsd.gov.hk/en/pee/BEC_2012.pdf (accessed: January, 30 2017).
- Goeritno, Arief, Ritzkal, Ayumi Johan, (2016), “Kinerja Prototipe Sistem Elektronis Berbasis Mikrokontroler Arduino Uno R3 Untuk Pemantauan Analogi Instalasi Listrik,” di *Jurnal Ilmiah SETRUM*, Volume 5, No.2, hlm. 94-99, Desember 2016.
- Linsley, Trevor, (2005), “*Basic Electrical Installation Work*”, Elsevier Ltd., pp. 45-49.
- Proteus2000, (1998), “*Proteus 2000 Operations Manual*”, E-MU Systems, Inc., pp. 131-164.
- Sopyandi, Arief Goeritno, Rakhmad Yatim, (2016), “Prototipe Sistem Pengontrolan Berbasis Payload Data Handling Berbantuan Mikrokontroler Untuk Instalasi Listrik Rumah Tangga”, *Prosiding SNTI FTI-Usakti V-2016*, hal. 331-337.
- The MCS Electronics Team, (2008), “*BASCOM-AVR User Manual Introduction*”, MCS Electronics, pp. 222-252.
- Zhou, Li, Junshe An, (2013), “Design of Payload Data Handling System for Satellites,” in *Proceeding 2013 Third International Conference on Instrumentation, Measurement, Computer, Communication and Control (IMCCC)*, Shenyang, China, 21-23 Sept. 2013, pp. 1217-1220. [Online]. Available: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6840660&isnumber=6840388> (accessed: January, 30 2017).